Locating Physical Interface Objects on Interactive Surfaces

Florian Block, Hans Gellersen, Mike Hazas
David Molyneaux, Nicolas Villar

Computing Department, Lancaster University
Infolab21, South Drive, Lancaster LA1 4WA, U.K.

Abstract: Pin&Play has enabled a new type of surface-based physical user interface, characterised by dynamic arrangement of interface objects on a surface area. Previous work has shown that this affords rapid re-arrangement of the spatial layout of interface objects, for example in adaptation to user preferences, but the Pin&Play system did not support tracking of object locations on the surface. In this paper, we investigate and compare two practical location techniques for interactive surfaces that are based on external sensing: detection of surface events using load sensors, and camera-based detection using object beacons.

1 Introduction

In a drive to support user interaction and applications beyond the desktop, a wide range of environment-based interface technologies are emerging. Among these, tangible interactive surfaces have received considerable attention, as they extend familiar physical media, such as whiteboards, notice boards and workbenches, with digital interaction [UI97, MSM^{+}99, JIPP02]. The Pin&Play project demonstrated a new type of surface-based system in which the surface is augmented as ad hoc network medium for interactive objects [vSG02]. This concept has been extended in the VoodooIO architecture for physical interfaces that afford dynamic re-arrangement of interface objects on interactive surface areas [VGRG06].

The Pin&Play infrastructure is based on interactive surfaces with embedded conductive layers that provide a power and data bus. Physical interface objects can be attached as nodes to the surface (using coaxial pin connectors), and will immediately be discovered on the network bus and registered as part of the interface and henceforth monitored to track user interaction. Like attachment, removal of objects is detected instantaneously. This enables applications, in which the physical composition and spatial layout of the interface can be changed at run time.

While insertion, manipulation and removal of objects are tracked by Pin&Play, the infrastructure does not support location of objects. However, knowledge of the spatial arrangement of objects would extend the range of tasks and applications for Pin&Play. In fact, many applications reported for interactive surfaces involve spatial tasks, in which position of objects on the surface, or relative to other objects, is used as meaningful input

[KLLL04]. In this paper we investigate the extension of Pin&Play with techniques for locating objects on the surface. We focus on practical methods that use external sensing, and describe the implementation and characterization of two alternative techniques, one based on pressure sensing and the other using vision.

2 Two Location Methods for Pin&Play

An interactive surface with embedded network can be designed to sense location using the same means that provide connectivity to nodes. For example, surface electrodes might be chosen and laid out with a suitable topology [HS02] or resistivity which allows a connected node to be accurately located. However, these methods typically increase the complexity of surface construction or require specialised materials. To facilitate practical manufacture and deployment of our interactive surfaces, we thus investigate two location methods which rely on external sensing systems. This section details the two methods, and then shows how they interoperate with other parts of the Pin&Play architecture.

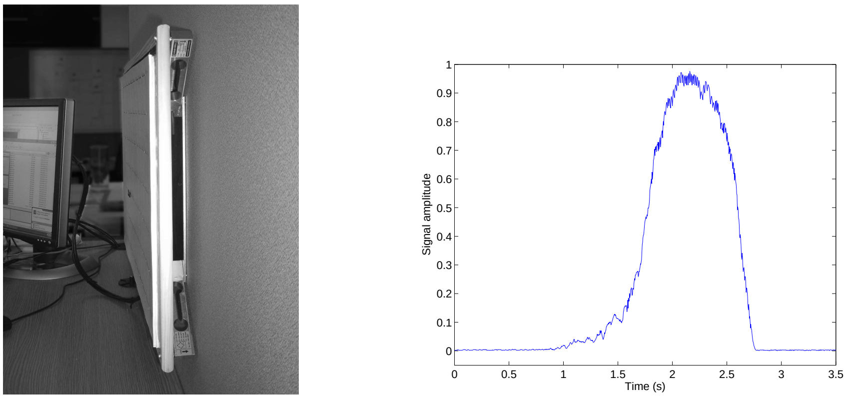
2.1 Locating surface events using pressure sensing

Using load sensors installed on the underside of a surface, it is possible to estimate the position of events which cause the force on the surface to change [SSv+02]. Assuming a rigid body mechanical model, the coordinates (u, v) of the point of applied force are linearly dependent on (a) the differential forces detected by the load sensors at the time of the event, and (b) the coordinates where each load sensor contacts the surface. More formally, differential forces F_i are measured from a set of N load sensors which have known locations (x_i, y_i), where $i = 1 \ldots N$. The coordinates (u, v) of the force event on the surface are defined as

$$u = \sum_{i=1}^{N} \left(\frac{F_i}{\sum_{j=1}^{N} F_j} \right) x_i \qquad\qquad v = \sum_{i=1}^{N} \left(\frac{F_i}{\sum_{j=1}^{N} F_j} \right) y_i \ . \tag{1}$$

In our implementation (Fig. 1a), the corners of a 60 × 40 cm board were affixed to four load sensors (manufactured by Bongshin Loadcell) mounted on a vertical surface. Supplied with a DC bias, each load sensor outputs a voltage proportional to the force applied; the maximum measureable load for each sensor is one kilogram. The load sensor outputs are passed through an instrumentation amplifier IC. Using an analogue-to-digital converter, a microcontroller samples each load signal at 300 Hz, and passes the sampled values to a workstation PC via a serial link.

Fig. 1b depicts a typical signal captured from a load sensor as a pin is pushed manually into the board. As the plot shows, the signals captured during a push event are quite distinct. A simple peak detection algorithm can be used to detect the event occurrence. The differential force F_i on a particular sensor is proportional to the average signal level at the top of



(a) Load sensor placement (b) Typical load signal when a pin is inserted

Abbildung 1: Load sensing for ad hoc interactive surfaces

the peak minus the average signal level before the peak occurred. Note that for the location computation (1), the resulting signal amplitude differential can be substituted directly for the differential force F_i, provided the load sensors have comparable sensitivites.

2.2 Camera-based location of interactive objects

Interface objects used as nodes in the Pin&Play system are fitted with a light emitting diode (LED) to provide visual feedback on insertion into a surface. With this output capability, the nodes may be polled to send out a beacon suitable for detection by a camera–based location system. The location system can estimate the position of the LED by using simple image processing to identify significant, highly localised changes in light intensity [KTV$^+$05]. A straightforward method of detecting intensity changes is to compute a simple difference between images taken before and after an LED has been turned on.

Localising an LED in the difference image yields coordinates referenced to the image itself. For example, the location might be expressed in pixel coordinates. To make the location result useful, it needs to be referenced to the surface on which the nodes are placed. The *projective mapping*, or homogenous transformation, is a robust technique which models image warping due to camera perspective; it can thus be used to convert between image coordinates and physical coordinates.

In a projective mapping, the points on one plane are projected through a single point in space onto another plane [Hec89]. As defined by a two-dimensional projective mapping, the relation between the image coordinates (u', v') and the coordinates on a plane (u, v) is

$$u = \frac{au' + bv' + c}{gu' + hv' + 1} \qquad\qquad v = \frac{du' + ev' + f}{gu' + hv' + 1} \quad . \tag{2}$$

To convert from image coordinates to physical coordinates for a particular camera setup, the eight coefficients (a through h) must be known. Given four or more unique points in image space and their corresponding points on the physical surface, the coefficients can be determined by reformulating the system of equations using matrix notation, and solving using Gaussian elimination or linear least squares. For details on computing the projective mapping coefficients and a survey of two-dimensional mapping techniques, the reader is referred to Heckbert [Hec89, Sect. 2.2].

Once the LED's position on the surface has been estimated, it can be related to the location of the node itself, using information about the size of the node and the exact placement of the LED on the node. It is also possible to compute the orientation of nodes equipped with two or more beacon LEDs.

The camera–based location system used for the experiments in this paper utilised an off-the-shelf webcam (a Trust SpaceCam 380) with 640×480 pixel resolution.[1] The webcam and an interactive surface were both connected to a workstation PC, which triggers nodes on the surface to beacon and performs image capture and location estimation. The webcam was placed about 110 cm from the surface, with its field of view covering an area approximately 85×64 cm. This yields a physical resolution of about 1.33 mm/pixel.

2.3 Integration in Pin&Play

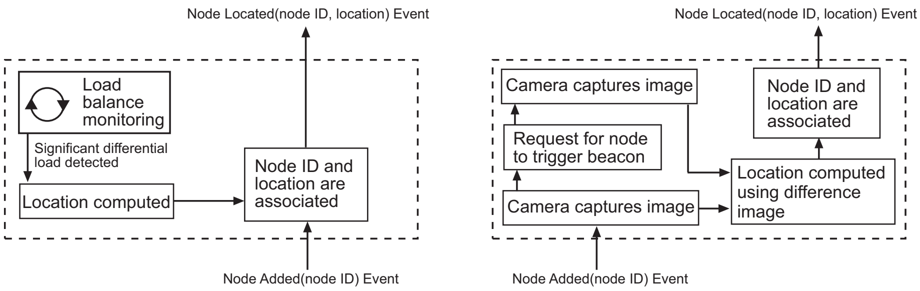
The two location techniques operate in slightly different ways in the context of the interactive surface system architecture. The load sensing method (Fig. 2a) computes a location when a force is applied to the surface, and then associates the result with the node's ID once the node registers with the system. In contrast, the camera-based method (Fig. 2b) waits for the new node to register, triggers the node to beacon, and then computes a location estimate. Note that both methods rely upon communication with the new node before returning a location result. Thus, false location events, such as those caused by bumps to the surface (for load sensing) or scene lighting changes (for camera-based detection), can often be identified and discarded.

3 Experiments and Analysis

A series of location experiments were conducted to aid in characterisation and comparison of the load sensing and camera–based location methods. This section describes these experiments, presents results quantifying the accuracy of the two systems, and then discusses and compares other aspects of the systems' performance.

Readings were taken at locations on a 10×7 grid on both surfaces. A grid spacing of 5 cm was used for the load sensing surface, whereas 9 cm was used for the larger surface covered

[1] The lenses used in inexpensive webcams often suffer from optical aberrations. Thus, a calibration procedure was performed for the camera, allowing the image processing software to partially compensate for lens effects.



(a) Load sensing (b) Camera detection

Abbildung 2: Node location methods in the context of Pin&Play

by the camera. As noted in Sect. 2, both systems require some knowledge, or *calibration*, which relates the sensor data to the physical surface.

For the load sensing system, the sensor locations (x_i, y_i) were surveyed manually with respect to the reference grid. To compute the projective mapping as defined by (2) for the camera-based system, two methods were used to find the four points in the image which correspond to the four corner points of the reference grid. First, a one-time *manual calibration* was carried out by a human to identify the pixel coordinates of the four corners in a captured image. Second, four nodes with beacon capability (i.e. a surface mount LED) were placed at the corner points on the grid, allowing an *autocalibration* to be performed by the system prior to gathering readings at each location. This autocalibration step involved flashing the LED at each corner node five times, which took approximately twenty seconds in total prior to each experiment. From the resulting difference images, the median pixel coordinates for each corner were used to compute the projective mapping.

For the load sensing tests, a tack-shaped node (1.5 cm in diameter) with a single coaxial pin connector was manually pushed into the surface fifty times at each of the seventy locations on the grid, for a total of 3500 location readings. For the camera tests, fifty readings were taken with a tack node placed at each of sixty-six points on the grid, for a total of 3300 location readings.[2]

3.1 Results

Figure 3 shows the error distributions for the two location methods. Although the camera-based implementation covers a larger surface area, it yields significantly lower error. In 95% of cases, the camera-based system was accurate to within 6 mm, compared to 18 mm for the load sensing system.

Much of the load sensing error can be attributed to the fact that the system is not perfectly

[2]The corners of the grid were not used as test locations for the camera-based system. Readings at these points yield artifically low error, since they are also used for autocalibration.

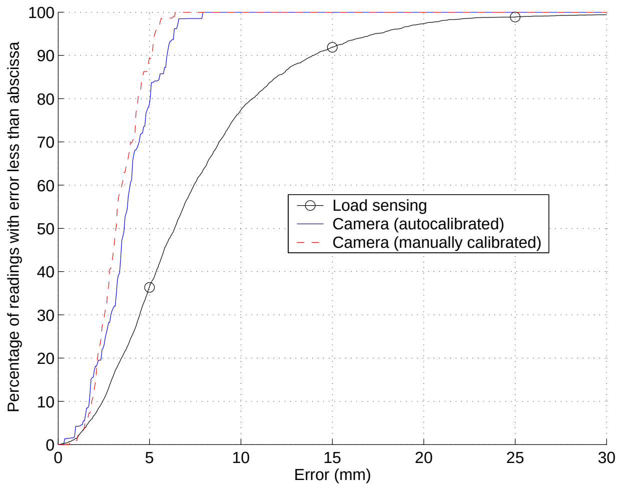


Abbildung 3: Location accuracy comparison

modeled by rigid body mechanics, as (1) assumes. In our implementation the surface frame and backing are made of inexpensive wood. When a node is pushed onto the surface, visible flexing of the surface occurs. Improved accuracies should be possible by providing added rigidity to the surface. This could be accomplished by using less pliant materials or by adding cross-support beams on the underside of the surface.

The results also indicate that manual calibration performs slightly better than autocalibration for the camera system. However, the improvement is about one millimetre at the 95% confidence level, which is comparable to the accuracy with which the reference grid was laid out on the surface.

3.2 Comparative Analysis

In this section we briefly analyse other aspects of the systems' performance in order to contrast the two location methods.

Detection Latency. The load sensing method provides location of a node within the time it takes the Pin&Play system to discover and register a node on the surface, and in this sense does not add further latency. The user perceived latency thus equals the time for network discovery of the node, typically under 400 ms, depending on the number of nodes connected. By contrast, node localisation with the camera-based system involves a significant latency in addition to node discovery. In our experiments with camera-based localisation, optimised for accuracy rather than speed, the user perceived latency was 2–3 s.

Calibration Needs. The load sensing method requires that the sensor locations be accurately surveyed. The camera method also requires a survey of at least four points in the image scene. For both systems, a similar amount of time was involved in carrying out this calibration manually. However, if four nodes are connected to the surface at known locations, the camera-based system also offers the option of an unassisted autocalibration without a significant loss in location accuracy.

Impact of Node Physical Attributes. In our implementation, both methods make assumptions about node physical attributes in order to compute a location. The load sensing method assumes that the centre of the force applied to the surface corresponds to the centre of the physical node. Likewise, the camera-based method currently assumes that beaconing LEDs are at the centre of the polled node. However, certain nodes may not have these attributes. For example, if a large node is pressed onto the load sensing surface, it is unlikely that the location of the largest force will correspond to the centre of the node. Or, if an LED is positioned away from a node's centre, then the location of the centre of the node will not be uniquely defined since the orientation of the node on the surface is not known.

However, for the case of large or unusually-shaped nodes, camera-based sensing may be the best solution. The exact placement of the LED on the node can be taken into account when interpreting the location estimate. A further enhancement would be to construct nodes using two or more LEDs, making orientation estimable in addition to increasing the reliability of location estimates.

Events Prohibiting Location Updates. Commonly in location systems, certain events occurring simultanously with sensor measurements can prohibit the system from returning valid location updates. The load sensing method can fail if a force differential occurs on another part of the surface while a new node is being added. The camera-based method can fail if there are people or objects occluding the beaconing LED at the time of image capture. However the camera-based method can periodically poll nodes in order to refresh the location estimates, whereas the load sensing method can only detect location while the node is being pressed onto the surface.

4 Conclusion

We have implemented and analysed two sensing methods for extension of Pin&Play interactive surface with the ability to locate objects as they become attached. Both methods, load-sensing and camera-based localisation, are very practical in terms of implementation and deployment, and provide a level of accuracy that would support disambiguation of objects and a wide range of spatial tasks. Neither of the methods requires any alteration of the core Pin&Play system, and can be added at relatively low cost.

The two methods have distinct advantages and disadvantages. Load-sensing allows for very fast localisation of nodes, however with only one try: if the node is not successfully located on insertion the system will not be able to obtain its location until it is removed and re-inserted. Camera-based sensing involves a significant latency but the system can locate nodes at any time following their insertion and is thus able to recover from initial localisation failure. The load-sensing method is well aligned with user interaction, as the act of inserting a node on the surface directly triggers localisation. In contrast, user interaction tends to obstruct visual node localisation with the camera-based method. However the camera-based method has the advantage that it can be extended to provide more information, for example to detect object shapes and object orientation in addition to their location. Finally, in terms of deployment, we assume that a camera can be easily aligned with a surface, but the load-sensing method has the advantage can be fully embedded with Pin&Play surfaces, practically as a single unit.

Literatur

[Hec89] Paul S. Heckbert. Fundamentals of Texture Mapping and Image Warping. Diplomarbeit, University of California, Berkeley, 1989.

[HS02] Frank Hoffmann und James Scott. Location of Mobile Devices Using Networked Surfaces. In *Proceedings of UbiComp: Ubiquitous Computing*, Seiten 281–298, Göteborg, Sweden, September 2002. Springer-Verlag.

[JIPP02] Robert Jacob, Hiroshi Ishii, Gian Pangaro und James Patten. A tangible interface for organizing information using a grid. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*, Seiten 339–346, Minneapolis, USA, April 2002.

[KLLL04] Scott R. Klemmer, Jack Li, James Lin und James A. Landay. Papier-Mache: toolkit support for tangible input. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, Seiten 399–406, Vienna, 2004. ACM Press.

[KTV+05] Yasue Kishino, Tsotomu Terada, Nicolas Villar, Hans Gellersen und Shojiro Nishio. Position Detection Mechanism using Camera Images for Pin and SPlay. In *Adjunct Proceedings of Ubicomp 2005*, Tokyo, Japan, 2005.

[MSM+99] Thomas P. Moran, Eric Saund, William Van Melle, Anuj U. Gujar, Kenneth P. Fishkin und Beverly L. Harrison. Design and technology for Collaborage: collaborative collages of information on physical walls. In *UIST '99: Proceedings of the 12th annual ACM symposium on User interface software and technology*, Seiten 197–206. ACM Press, 1999.

[SSv+02] Albrecht Schmidt, Martin Strohbach, Kristof van Laerhoven, Adrian Friday und Hans-W. Gellersen. Context Acquisition Based on Load Sensing. In *Proceedings of UbiComp: Ubiquitous Computing*, Seiten 333–350, Göteborg, Sweden, Oktober 2002. Springer-Verlag.

[UI97] Brygg Ullmer und Hiroshi Ishii. The metaDESK: Models and Prototypes for Tangible User Interfaces. In *Proceedings of UIST '97: ACM Symposium on User Interface Software and Technology*, Seiten 223–232, Oktober 1997.

[VGRG06] Nicolas Villar, Kiel Gilleade, Devina Ramduny-Ellis und Hans Gellersen. VoodooIO Gaming Kit: A real-time adaptable gaming controller. In *ACE '06: Proceedings of Intl. Conference on Advances in Computer Entertainment Technology*. ACM Press, 2006.

[vSG02] Kristof van Laerhoven, Albrecht Schmidt und Hans-W. Gellersen. Pin and Play: Networking Objects through Pins. In *Proceedings of the International Conference on Ubiquitous Computing (Ubicomp)*, Seiten 219–229, Göteborg, Sweden, September 2002.